



SmartModel Library Release Notes

To search the entire manual set, press this toolbar button.
For help, refer to [intro.pdf](#).



September 2001

Copyright © 2001 Synopsys, Inc.

All rights reserved.

Printed in USA.

Information in this document is subject to change without notice.

Synopsys and the Synopsys logo are registered trademarks of Synopsys, Inc. For a list of Synopsys trademarks, refer to this web page:

<http://www.synopsys.com/copyright.html>

All company and product names are trademarks or registered trademarks of their respective owners.

Contents

Release Notes	5
Introduction	5
What's New?	5
swiftpli Shared Libraries for Verilog	6
SmartModels on Linux	6
Flexmodel Enhancements	6
Simulator Configuration Guide for Synopsys Models	7
SystemC/SWIFT Support	7
New Tool, cnvrt2mif	7
Using PCI/PCI-X FlexModels	7
Install the Entire Set of PCI/PCI-X FlexModels	8
PCI/PCI-X Documentation	8
PCI/PCI-X FlexModel Licensing	8
Ensuring Compatibility With PCI/PCI-X FlexModels	9
FlexModel Issues and Workarounds	13
SmartModel Issues and Workarounds	14
Synopsys Models on Scirocco	14
SourceModel Users—Need Separate Install Directories	14
Synopsys Models with VCS on HP-UX 11.0	15
VSS Simulator Users—Faulty Memory Windows	15
NT Issues and Workarounds	16
Getting Help	17
The Synopsys Website	17
Synopsys Common Licensing (SCL) Document Set	18
Comments?	18

Release Notes

Introduction

This document contains information about the following topics:

- [“What’s New?” on page 5](#)
- [“Using PCI/PCI-X FlexModels” on page 7](#)
- [“FlexModel Issues and Workarounds” on page 13](#)
- [“SmartModel Issues and Workarounds” on page 14](#)
- [“NT Issues and Workarounds” on page 16](#)
- [“Getting Help” on page 17](#)

For general information about SmartModel Library documentation, or to navigate to a different online document, refer to the [Guide to SmartModel Documentation](#). For information about which platforms, operating systems, and simulators support SmartModels and FlexModels, refer to [SmartModel Library Supported Simulators and Platforms](#).

What’s New?

These release notes are frequently updated to keep you informed about important changes in the SmartModel software or alert you to issues and workarounds that you should know about. If you want to focus on just the information that is new or noteworthy, use this section as your guide.

- [“swiftpli Shared Libraries for Verilog” on page 6](#)
- [“SmartModels on Linux” on page 6](#)
- [“Flexmodel Enhancements” on page 6](#)
- [“Simulator Configuration Guide for Synopsys Models” on page 7](#)

- [“SystemC/SWIFT Support” on page 7](#)
- [“New Tool, cnvrt2mif” on page 7](#)

swiftpli Shared Libraries for Verilog

Synopsys now ships the LMTV PLI application and other interface code in the form of swiftpli shared libraries (.so, .sl, or .dll) for Verilog-XL, NC-Verilog, and MTI Verilog on all supported platforms. With these new shared libraries there is no need to build a new Verilog executable or a libpli. Just point your load path variable to the platform-specific directory in \$LMC_HOME and use a switch on the simulator invocation line to specify the swiftpli. Changes to the LMTV interface will now get picked up automatically whenever you update your model installation. For more information, refer to the [Simulator Configuration Guide for Synopsys Models](#).

SmartModels on Linux

SmartModels (including FlexModels) are now available on the Red Hat Linux 6.1 platform.

Flexmodel Enhancements

FlexModels were recently enhanced with the following features:

- **Reset**—FlexModels now support resetting a simulation back to time zero during a simulation run.
- **Uncoupled Mode**—FlexModels now support multiple command streams that operate independently in a C testbench.
- **Multiple Command Streams**—A single C testbench can now provide commands to more than one FlexModel or multiple instances of the same FlexModel.
- **Linux Support**—FlexModels are now supported on Red Hat Linux 6.1.
- **AIX Support**—FlexModels are now supported on AIX 4.3.2.
- **New interrupt commands**—the commands, FLEX_SET_VALUE, FLEX_GET_VALUE, and FLEX_WAIT_ON_NODE, allow you to prompt the simulator for values on individual nets from within a C program. You can set, get, or wait on the value of any net in a design.

For information about using multiple command streams and uncoupled mode with C testbenches, refer to the [FlexModel User's Manual](#).

Simulator Configuration Guide for Synopsys Models

The [Simulator Configuration Guide for Synopsys Models](#) is the latest addition to the SmartModel Library documentation set. This guide explains how to integrate SmartModels, FlexModels, MemPro models, and hardware models with the most widely used simulators. It also explains how to integrate SmartModels and FlexModels on all supported SWIFT simulators. Along with this enhancement, the *SWIFT Usage Notes for QuickSim II and Verilog-XL* has been retired. All of the information formerly in that manual is now contained in the new guide.

SystemC/SWIFT Support

Synopsis provides a SystemC/SWIFT interface that supports Flex Models. SystemC is a C++ class library used for creating cycle-accurate models of software algorithms, hardware architecture, and interfaces for System-on-Chip (SoC) and system-level designs. As part of its class library, SystemC provides a cycle simulation environment, is designed to work with event-driven logic simulators, and provides extensive support for modeling device timing accurately. For more details see the [SmartModel Products Application Notes Manual](#).

New Tool, cnvt2mif

A new tool, cnvt2mif, replaces mi_trans for converting memory data files to the memory image file (MIF) format. Using cnvt2mif, you can optionally specify the number, name, and bitwidth of output files and perform endian conversion. In addition, you can select data from within the input file by specifying an address range for the conversion. You can also specify a base address for indexed addressing, and control the verbosity and extent of message display. cnvt2mif reads hexadecimal digits in either upper or lower case. You should use cnvt2mif for new designs, but mi_trans is still supported for backward compatibility. For more details see the [SmartModel Library User's Manual](#).

Using PCI/PCI-X FlexModels

The PCI FlexModels have been enhanced to support the PCI-X Addendum to the PCI Specification, Version 1.0. You can use PCI/PCI-X FlexModels in either PCI-X mode or conventional PCI mode. In PCI-X mode, the PCI/PCI-X FlexModels support the 66 MHz and 133 MHz timing versions as described in the PCI-X Addendum.

The PCI/PCI-X FlexModel set includes PCI and PCI-X system testbenches, plus the `pcimaster_fx`, `pcislave_fx`, and `pcimonitor_fx` models with the MDL version numbers shown below:

- **pcimaster_fx**: MDL version 01018 (1.18) or higher
- **pcislave_fx**: MDL version 01016 (1.16) or higher
- **pcimonitor_fx**: MDL version 01017 (1.17) or higher

For PCI FlexModel Test Suite users, the set also includes an updated PCI Test Suite (MDL version 01006 or above) that supports the PCI/PCI-X FlexModels.

The individual PCI/PCI-X FlexModels work together as a set to help you verify designs developed for use with PCI and PCI-X bus interface standards. Because of this, the PCI/PCI-X bus interface models are different from other FlexModels in several important ways, as described in the following sections.

Install the Entire Set of PCI/PCI-X FlexModels

The PCI/PCI-X FlexModels are tested in combination, so we recommend that you reinstall all models each time you update one or more in a set to make sure you stay with a tested and supportable combination. If you are a PCI FlexModel Test Suite user, you also need to reinstall the PCI Test Suite every time you update the model set.

PCI/PCI-X Documentation

Model-specific information for the three PCI/PCI-X FlexModels is documented in the *PCI/PCI-X FlexModel User's Manual*. This manual ships with each of the individual PCI/PCI-X FlexModels (`pcimaster_fx`, `pcislave_fx`, and `pcimonitor_fx`). The model version and history information provided in the addendum to this manual is specific to the individual models. You can view datasheets for installed models using the browser tool (`$LMC_HOME/bin/sl_browser`). To see datasheets for the latest available model versions, use the Model Directory on the Web:

<http://www.synopsys.com/products/lm/modelDir.html>

PCI/PCI-X FlexModel Licensing

You need a specific type of group model license (SIMMODEL-SW-pci_fx) to authorize all three of the PCI/PCI-X FlexModels. Contact your Synopsys sales rep for assistance in obtaining group licenses.

Ensuring Compatibility With PCI/PCI-X FlexModels

When using the PCI/PCI-X FlexModels, you should take note of the following compatibility issues:

- [“Compatibility Within the PCI/PCI-X FlexModel Set” on page 9](#)
- [“Compatibility With Existing Testbenches” on page 9](#)
- [“Compatibility With the PCI FlexModel Test Suite” on page 12](#)
- [“Converting Existing PCI SourceModel Testbenches to FlexModel” on page 12](#)

Compatibility Within the PCI/PCI-X FlexModel Set

It is especially important that you reinstall all models in the PCI/PCI-X FlexModel set each time you update one or more PCI FlexModels. This ensures that you are using a tested and supportable combination. If you are a PCI FlexModel Test Suite user, this applies to the FlexModel Test Suite as well.

Compatibility With Existing Testbenches

With the PCI/PCI-X FlexModels, you can use the model in either conventional PCI mode or PCI-X mode. In conventional PCI mode, PCI models are backward-compatible with the exception of some pin changes and a change to the TimingVersion generic/defparameter. These exceptions require you to use one of two methods to make your existing testbench fully compatible with the PCI/PCI-X FlexModels:

- Method 1: Modify your existing testbench (recommended)
- Method 2: Install the Synopsys-supplied compatibility files.

Method 1 lets you use PCI-X mode and all other enhancements. This is the recommended method. The changes required by this method need only be done once, making future upgrades easier and version control simpler. This is the method you need to use if you are running the updated PCI FlexModel Test Suite (MDL version 01006 or higher).

Method 2 lets you use the PCI/PCI-X FlexModels with your existing testbench, but does not allow you to use the models in PCI-X mode. If you use Method 2, you must reinstall the compatibility files every time you upgrade your PCI models. This method makes the PCI/PCI-X FlexModels incompatible with the updated PCI FlexModel Test Suite (MDL version 01006 or above).

Method 1: Modifying Your Existing Testbench

To change your existing testbench to make it fully compatible with the PCI/PCI-X FlexModels, follow these steps:

1. Change the TimingVersion generic/defparameter for each instance.

You can use 33 or 66MHz timing in conventional PCI mode, and 66 or 133MHz timing in PCI-X mode; therefore, there are two new settings for the TimingVersion generic/defparameter: “conventional” and “pcix”. To use 33 or 66MHz timing in conventional PCI mode, you must change the TimingVersion generic/defparameter for each model instance to “conventional”.

2. Modify your pcimaster_fx instantiation, if necessary, to make the pack64nn and pserrnn pins bidirectional.

When PCI-X mode was added to the pcimaster_fx, it was necessary to change two pins (pack64nn and pserrnn) from input to bidirectional. This may require you to modify your instantiation of the pcimaster_fx.

3. Modify your pcislave_fx instantiation, if necessary, to include new pins or new pin directions.

When PCI-X mode was added to the pcislave_fx, it was necessary to add some pins and change the direction of others. This may require you to modify your instantiation of the pcislave_fx. [Table 1](#) lists the pins that are new or have changed.

Table 1: New and Changed Pins for the pcislave_fx

Device Pin Name	Direction	Model Pin Name
C/BE#[3:0]	In/out	PCISLAVE_PCXBENN_BUS
C/BE#[7:4]	In/out	PCISLAVE_PBENN_BUS
GNT# (new pin)	In	PCISLAVE_PGNTNN_PIN
IRDY#	In/out	PCISLAVE_PIRDYNN_PIN
LOCK#	In/out	PCISLAVE_PLOCKNN_PIN
REQ# (new pin)	In/out	PCISLAVE_PREQNN_PIN
REQ64#	In/out	PCISLAVE_PREQ64NN_PIN
FRAME#	In/out	PCISLAVE_PFRAMENN_PIN
ACK64#	In/out	PCISLAVE_PACK64NN_PIN
DEVSEL#	In/out	PCISLAVE_PDEVSELNN_PIN

Table 1: New and Changed Pins for the pcislave_fx (Continued)

Device Pin Name	Direction	Model Pin Name
PERR#	In/out	PCISLAVE_PPERRNN_PIN
SERR#	In/out	PCISLAVE_PSERRNN_PIN

Method 2: Installing the Compatibility Files

The easiest way to install the compatibility files is to use the flexm_setup script to set up a work directory that contains all the package files, as shown below. For details on using the flexm_setup script, see the [FlexModel User's Manual](#).

1. Run flexm_setup:

To copy all the files required to run your existing testbench, including the compatibility files, run the following commands:

```
% mkdir workdir
% $IMC_HOME/bin/flexm_setup -dir workdir pcislave_fx
% $IMC_HOME/bin/flexm_setup -dir workdir pcimaster_fx
% $IMC_HOME/bin/flexm_setup -dir workdir pcimonitor_fx
```

2. Copy the compatibility files:

This procedure works as shown only after you run flexm_setup as shown in Step 1.

For VHDL:

```
% cd workdir/examples/vhdl
% cp pcislave_old.vhd pcislave.vhd
% cp pcimaster_old.vhd pcimaster.vhd
% cp pcimonitor_old.vhd pcimonitor.vhd

% cd ../../src/vhdl
% cp pcislave_pkg_old.vhd pcislave_pkg.vhd
% cp pcimaster_pkg_old.vhd pcimaster_pkg.vhd
% cp pcimonitor_pkg_old.vhd pcimonitor_pkg.vhd
```

For Verilog:

```
% cd workdir/examples/verilog
% cp pcislave_old.v pcislave.v
% cp pcimaster_old.v pcimaster.v
% cp pcimonitor_old.v pcimonitor.v
```

3. Compile VHDL packages for simulation (VHDL only):

For VHDL only, you must make sure that all packages are compiled for simulation. The following is an example for running the MTI simulator using the Solaris platform. If you are running on an HP platform, the commands are slightly different.

```
% cp $LMC_HOME/lib/sun4Solaris.lib/slm_mti.so .

% vlib mti_work
% vmap slm_lib ./mti_work
% vcom -work slm_lib $LMC_HOME/sim/mti/src/slm_hdlc.vhd
% vcom -work slm_lib $LMC_HOME/sim/mti/src/flexmodel_pkg.vhd

% vcom -work slm_lib workdir/src/vhdl/pcislave_user_pkg.vhd
% vcom -work slm_lib workdir/src/vhdl/pcislave_pkg.vhd
% vcom -work slm_lib workdir/src/vhdl/pcimonitor_user_pkg.vhd
% vcom -work slm_lib workdir/src/vhdl/pcimonitor_pkg.vhd
% vcom -work slm_lib workdir/src/vhdl/pcimaster_user_pkg.vhd
% vcom -work slm_lib workdir/src/vhdl/pcimaster_pkg.vhd

% vcom -work slm_lib workdir/examples/vhdl/pcislave_fx_mti.vhd
% vcom -work slm_lib workdir/examples/vhdl/pcimaster_fx_mti.vhd
% vcom -work slm_lib workdir/examples/vhdl/pcimonitor_fx_mti.vhd

% vcom -work slm_lib workdir/examples/vhdl/pcisys_fx_comp.vhd

% vcom -work slm_lib workdir/examples/vhdl/pcimonitor.vhd
% vcom -work slm_lib workdir/examples/vhdl/pcislave.vhd
% vcom -work slm_lib workdir/examples/vhdl/pcimaster.vhd
```

Compatibility With the PCI FlexModel Test Suite

To ensure full compatibility between the PCI/PCI-X FlexModels and the PCI FlexModel Test Suite, you must install the updated PCI Test Suite (MDL version 01006 or above) that comes with your PCI/PCI-X FlexModel set.

Converting Existing PCI SourceModel Testbenches to FlexModel

Synopsys provides a translation script to help you migrate existing PCI SourceModel testbenches to the FlexModel environment. This script converts SourceModel commands into equivalent FlexModel commands. For information on converting PCI SourceModel testbenches to the FlexModel environment, see Appendix C in the *PCI FlexModel User's Manual*.

FlexModel Issues and Workarounds

Following are known issues with the FlexModel software, along with suggested workarounds:

- When using multiple instances of a FlexModel within one or more top level Verilog testbenches (VCS, Verilog-XL,...) you may see the message:

```
Error: undefined symbol "flex_<cmd name>" (<testbench> line <number>)
```

To work around this error

- a. Add the line

```
`undef FLEXMODEL_CMDS_INC
```

before the line that reads

```
`include model_pkg.inc
```

- b. On the simulator invocation line add the multi-instance specification to your invocation.

```
+define+flex_multi_inst
```

- The *model_set_pin* commands do not work in timing mode. The observed behavior is that certain pins do not set. We recommended that you do not use this command when timing mode is turned on.
- The Veribest (Verilog and VHDL) model wrapper generator tools on Intel NT do not work with FlexModels.
- In VSS, interrupts are not supported within an HDL testbench. The workaround is to use C Command Mode interrupts.
- In VSS, the *flex_clear_queue* command causes the simulator to output the following error message:

```
"Error: internal memory allocation error."
```

You can ignore this message because the functionality of the command is correct.

- Do not use multiple HDL command streams to control a single FlexModel instance. This may cause unpredictable behavior.
- If you get an error message similar to the following example, you need to recompile your C testbench. (This message appears as an exception in the transcript window.)

```
C interface versions of flexmodel_pkg(1) & fastm_cmdcore(2) do not  
match
```

This error occurs when your testbench has been compiled with an earlier version of flexmodel_pkg and you have updated your LMC_HOME tree with the latest flexmodel_pkg and Command Core. For information on how to recompile a C testbench, see the [FlexModel User's Manual](#).

SmartModel Issues and Workarounds

Following are known issues with the SmartModel software, along with suggested workarounds:

- [Synopsys Models on Scirocco](#)
- [SourceModel Users—Need Separate Install Directories](#)
- [Synopsys Models with VCS on HP-UX 11.0](#)
- [VSS Simulator Users—Faulty Memory Windows](#)
- [NT Issues and Workarounds](#)



Attention

With the Mentor Graphics D.1 release, support for the SWIFT integration on QuickSim II transferred from Synopsys to Mentor Graphics. If you are using a D.1 or higher release of QuickSim II, refer to the Mentor Graphics documentation for information about using the SWIFT interface with their software. If you are using a Mentor Graphics release prior to D.1, refer to the [Simulator Configuration Guide for Synopsys Models](#).

Synopsys Models on Scirocco

With Scirocco version 2000.02, you must use the -debug_all switch on the simulator invocation line. This switch is not needed with version 2000.06 and above.

Also, versions 2000.02 and 2000.06 of Scirocco generate faulty memory window arrays for SmartModels. For information on the workaround, refer to [“VSS Simulator Users—Faulty Memory Windows” on page 15](#).

SourceModel Users—Need Separate Install Directories

SmartModel and SourceModel products must have separate install directories. Do not install SmartModel products in the same install directory as SourceModel products. If you are using both R9 SourceModels (VHDL or Verilog) and SmartModels, set the LMG_HDL_HOME environment variable to the SourceModel install directory, and the LMC_HOME environment variable to the SmartModel install directory.

Synopsys Models with VCS on HP-UX 11.0

Customers using Synopsys models with VCS 5.2.1 or VCS 6.0 on the HP-UX 11.0 platform should contact

vcs_support@synopsys.com

or call 1-800-VERILOG for assistance.

VSS Simulator Users—Faulty Memory Windows

VSS can generate incorrect SmartModel memory window arrays. For certain versions of VSS to run properly with SmartModels it is necessary to run a script that comments out the faulty signal memory declarations. This script fixes the models so that VSS can function but does not make the memory windows feature available with VSS.



Note

This problem is known to occur in VSS versions prior to 1997.08 and again in VSS 1999.05 (SIM1).

If VSS crashes without generating any error messages and you have one or more SmartModel memory models in your design, it could be that you have these faulty memory windows declarations in your entities.vhd file. To solve this problem, use the following procedure.

Revised VSS Installation

VSS VHDL simulator users must add a step to the installation, as follows:

1. Install the SmartModel Library according to the directions in the *SmartModel Library Installation Guide*.
2. Follow the VSS directions for installing the SmartModel Library up through the point of running the create_smartmodel_lib command.
3. Execute the Perl script as shown below:

```
% memory_array
```

The script edits the file “entities.vhd,” commenting out the memory windows declarations.

4. If you have installed the file “entities.vhd” in any location other than the default, use that path as the argument for this command:

```
% memory_array [path]
```

5. The default location is \$LMC_HOME/synopsys/entities.vhd.

6. Recompile the file “entities.vhd” as shown below:

```
% create_smartmodel_lib -analyze
```

NT Issues and Workarounds

The following issues and workarounds apply only to the Intel NT platform:

- There is no longer a win32.lib directory at %LMC_HOME%\lib. Instead, there is a pcnt.lib directory. If your EDA software looks for the libswift.dll in the win32.lib it will not find the directory. This will cause the software to issue an error message about being unable to load SWIFT.

To solve this problem, create a win32.lib directory at %LMC_HOME%\lib and copy over the libswift.dll file from the pcnt.lib or alphant.lib directory.

- The SmartModel Library save and restore feature for SmartCircuit models is currently not functional on NT.
- ASCII readme files have .txt extensions. Some Windows NT installations are set up by default to associate such files with the Microsoft Notepad tool, which does not display the files correctly. If you have this problem, try using WordPad to read the ASCII files.
- The modeldiff tool is not supported on NT.
- Incremental Installs—When you use the Admin tool to install additional or revised models into an existing SmartModel Library installation (%LMC_HOME%) the tool may generate error messages about being unable to update certain files because they are in use. For example, if you have the license server running, the Admin tool will be unable to update lmgrd because it is in use. If you experience this type of problem, the workaround is to:
 - Make sure there are no users on your network that are simulating with SmartModel Library models
 - Shut down the license server
 - Run the Admin tool off the CD-ROM to perform the incremental install
- Reset/Restart with Flexmodels—The reset/restart feature of FlexModels is not supported in the NT environment.
- For the VCS simulator on the NT platform, HDL interrupts are supported in version 5.1 and higher only.
- For the VCS simulator on the NT platform, commands flex_set_value, flex_get_value and flex_wait_on_node are supported in version 5.1 and higher only.

Getting Help

If you have a question while using Synopsys products, use the following resources:

1. Start with the available product documentation installed on your network or located at the root level of your Synopsys CD-ROM. Every documentation set contains overview information in the [intro.pdf](#) file.

Additional Synopsys documentation is available at this URL:

<http://www.synopsys.com/products/lm/doc>

Datasheets for models are available using the Model Directory:

<http://www.synopsys.com/products/lm/modelDir.html>

2. Visit the online Support Center at this URL:

<http://www.synopsys.com/support/lm/support.html>

This site gives you access to the following resources:

- SOLV-IT!, the Synopsys automated problem resolution system
- product-specific FAQs (frequently-asked questions)
- lists of supported simulators and platforms
- the ability to open a support help call
- the ability to submit a delivery request for some product lines

3. If you still have questions, call the Support Center:

North American customers:

Call the Synopsys EagleI and Logic Modeling Products Support Center hotline at 1-800-445-1888 (or 1-503-748-6920) from 6:30 AM to 5 PM Pacific Time, Monday through Friday.

International customers:

Call your local sales office.

The Synopsys Website

General information about Synopsys and its products is available at this URL:

<http://www.synopsys.com>

Synopsys Common Licensing (SCL) Document Set

Synopsys common licensing (SCL) software is delivered on a CD that is separate from the tools that use this software to authorize their use. The SCL documentation set includes the following publications, which are located in (root)/docs/scl on the SCL CD and also available on the Synopsys FTP server (<ftp://ftp.synopsys.com>):

- *Licensing QuickStart*—(142K PDF file)
This booklet provides instructions for obtaining an electronic copy of your license key file and for installing and configuring SCL on UNIX and Windows NT.
- *Licensing Installation and Administration Guide*—(2.08M PDF file)
This guide provides information about installation and configuration, key concepts, examples of license key files, migration to SCL, maintenance, and troubleshooting.

You can find general SCL information on the Web at:

<http://www.synopsys.com/keys>

Comments?

To report errors or make suggestions, please send e-mail to:

doc@synopsys.com

To report an error that occurs on a specific page, select the entire page (including headers and footers), and copy to the buffer. Then paste the buffer to the body of your e-mail message. This will provide us with information to identify the source of the problem.